Ansible
Basics & Best practices

Cette présentation contient les bases et les bonnes pratiques pour utiliser Ansible

Université Gustave Eiffel

Thibault SAUSSAC - saussact@gmail.com

Agenda

1. Overview
2. Architecture
3. Features
4. Installation
5. Structure
6. Inventory File
7. Modules
8. Playbooks
9. Support and Licensing
10. Ansible Tower
11. Questions/Answers

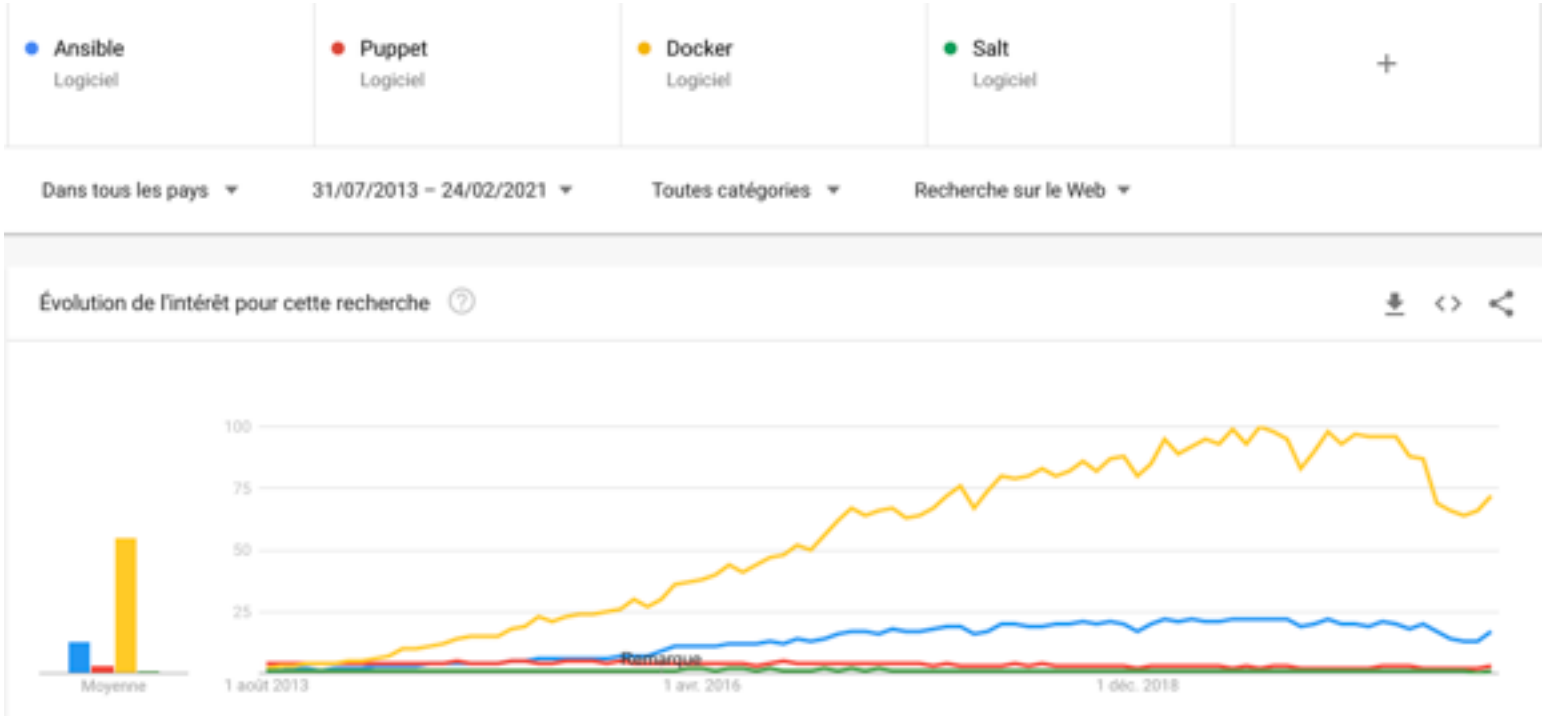
Overview

- Un outil de déploiement automatique qui permet d'automatiser le déploiements de systèmes distants et les déploiement d'applications.
- Utilisation d'une syntaxe en YAML pour l'écriture des « tasks » très proche de l'anglais natif
- Le model Ansible permet de décrire comment les systems de l'infrastructure interagissent plutôt que de simplement gérer un système.

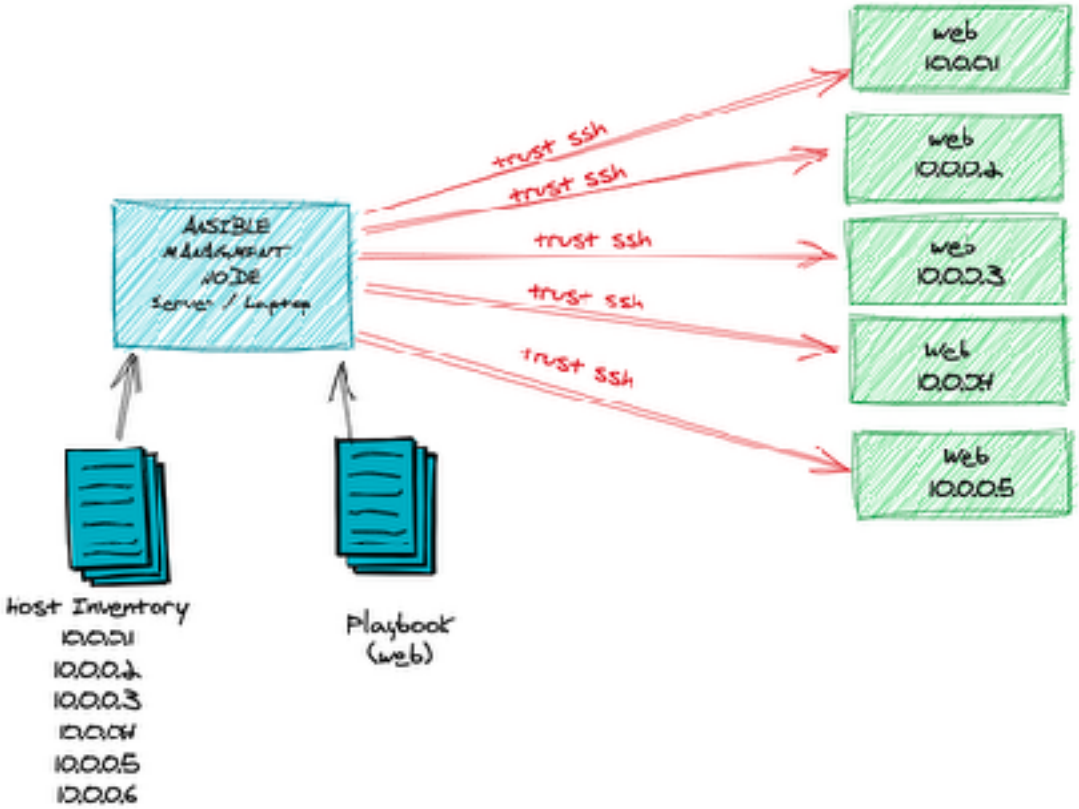
Overview - Pourquoi un outil de déploiement ?

- Avant les outils d'automatisation se faisait avec des scripts shell/ PowerShell ou tout autre langage de script.
  - Pas idempotent
  - Pas Robuste
  - Tout le monde tourne leur propre

Overview – Les concurrents de Ansible.



Ansible - Architecture



Ansible - Features

- Il y a beaucoup d'outil d'automatisation disponible : Chef, Puppet, Salt & Ansible.
- Basé sur le SSH, ce qui permet de ne pas avoir a installer d'agent sur chaque noeud distant.
- Le Yaml est assez proche du langage naturel.
- Utilise Python qui est aujourd'hui omniprésent sur toutes les distribs.
- La structure des playbook est simple et clair.
- **Ansible Galaxy** est le repo central d'Ansible, il permet rechercher, réutiliser et partager du code Ansible.
- SSH / SSH2 / Vault : permet d'avoir sécurité excellente.

Features - Ce qui fait défaut à Ansible?

- Il n'y a pas d'UI
- Les IDE ne sont pas très performant pour faire de l'introspection : IntelliJ, VisualCode, Eclipse
- Le support sur Window est impossible, la machine 'controler' peut uniquement être un OS unix.

Ansible - Installation

- CentOS/Fedora
  - sudo yum install epel-release
  - sudo yum install ansible
- MacOS
  - brew install ansible
- Ubuntu
  - sudo apt-get install software-properties-common
  - sudo apt-add-repository ppa:ansible/ansible
  - sudo apt-get update
  - sudo apt-get install ansible

Ansible Layout

- /etc/ansible — The main configuration folder which encompasses all Ansible config
- /etc/ansible/hosts — This file holds information for the hosts/and host groups you will configure
- /etc/ansible/ansible.cfg — The main configuration file for Ansible
- /etc/ansible/roles — This folder allows you to create folders for each server role, web/app/db, etc.

Ansible - Inventory

- Liste tous les « host » que Ansible Administre
- Vous pouvez définir des « groupes »
- Par défaut : /etc/ansible/hosts.

- Demo :)

Ansible - Playbooks

- Ecris en YAML
- Composé d'un ou plusieurs « play »
- Permet de déployer sur plusieurs machine
- Executer un playbook :
  - ansible-playbook site.yml
- Demo :)

```yaml
- name : simple Loclhost pwd
  hosts: web
  tasks:
    - name : ping
      command : pwd
      register: result
    - debug : msg="{{ result.stdout }}"
```

Ansible - Modules

- Permet de factoriser
- Utiliser le plus possible les modules existants.
- Et dans le code :

```
- name: Upgrade main.yml packages, excluding kernel & foo related packages
  yum:
    name: '*'
    state: latest
    exclude: kernel*,foo*
```

Ansible - Variable

- Défini directement dans les playbook
- Dans les templates Jinga
- Dans en variable externe
- Inventaire

- Demo
-

Ansible - Conditions and Loops

```
-
  name : httpd Install
  hosts: all
  tasks:
    - name: install httpd on centos
      yum:
        name: httpd
        state: present
      when: {{ condition }}
  name : httpd Install on redhat
  hosts: all
  tasks:
    - name: install httpd on centos
      apt:
        name: httpd
        state: present
      when: {{ condition }}
```

When condition

```
name : httpd Install
hosts: all
tasks:
  - name: install httpd on centos
    yum:
      name: httpd
      state: present
    when: ansible_os_family == "centos" or  ansible_os_family == "suse"
  - name: install httpd on redhat
    apt:
      name: httpd
      state: present
    when: ansible_os_family == "debian" and ansible_os_family == "ubuntu"
```

Ansible - Conditions and Loops

```
- name: Create WordPress db
  mysql_db:
    name: {{item}}
    state: present
  with_items :
    - {{wp_db_name}}
    - {{wp_db_name2}}
```

Ansible - Roles

TASKS - contains the main list of tasks to be executed by the role.

HANDLERS - contains handlers, which may be used by this role or even anywhere outside this role.

DEFAULTS - default variables for the role

VARS - other variables for the role (high precedence)

FILES - contains files which can be deployed via this role.

TEMPLATES - contains templates which can be deployed via this role.

META - defines some meta data for this role.

```
site.yml
roles/
   common/
     files/
     templates/
     tasks/
     handlers/
     vars/
     defaults/
     meta/
   webservers/
     files/
     templates/
     tasks/
     handlers/
     vars/
     defaults/
     meta/
```

Using Roles in playbook:

```
- hosts: db
  remote_user: vagrant
  roles:
    - common
    - mysql
```

Ansible - quelques commandes

- syntax check

  ansible-playbook playbook.yml --syntax-check

- Verbosity Level

  ansible-playbook playbook.yml –v

- Dry run

  ansible-playbook playbook.yml --check